JOURNAL OF COMPUTATIONAL PHYSICS **60**, 437–446 (1985)

An Analytic Algorithm for Decoupling and Integrating Systems of Nonlinear Partial Differential Equations

THOMAS WOLF

Sektion Physik, Friedrich-Schiller-Universität Jena, 6900 Jena, West Germany

Received March 23, 1984; revised September 26, 1984

An analytic method is presented which by taking advantage of integrability conditions decides if a system of equations, in which the functions and their partial derivatives have only positive integer exponents, allows solutions and leads to new equations of lower order. The method is especially striking if the equations are overdetermined and can be implemented on a computer. An application to the full vacuum field equations of general relativity in the presence of a Killing vector leads to the known formulation with potentials. © 1985 Academic Press, Inc.

1. INTRODUCTION

The motivation for developing an algorithm which decides if an overdetermined system of two equations for one function f has any solutions is to find new exact solutions of a system of field equations. Because these equations were too complex to be solved in general, additional assumptions had to be made which were strong enough to calculate the remaining free functions and were weak enough not to exclude all interesting cases and possible new solutions. The method considered fills in this gap because additional constraints in the form of differential equations need not have such a simple form as $f(x) = \text{const}$ or $f(x, y)_{,x} = 0$ to determine whether the system + additional constraint has any solution. To determine if there are solutions the algorithm describes how to generate a new equation which is in some sense easier than the most difficult one of the system (system + add. constr.). So this most difficult one is replaced by the new easier equation and the next easier equation is produced which replaces the previous equation and so on.

What shall be meant here by easy or difficult? If the system consists of ordinary differential equations then the degree of difficulty of an equation depends on the highest-order derivative of f; if they are equal in both equations then the highest power of the highest derivatives is decisive. If more than one variable occurs then their order must be defined, e.g. (y, x). So one equation is more difficult if it has higher y-derivatives or if their highest y-derivatives are equal and have greater x-derivatives of the highest y-derivatives. If they are also equal then the highest power

0021-9991/85 $3.00

of the highest x-derivative of the highest y-dervative is decisive. As an example, in the following line the terms become easier from the left to the right:

$$f_{,yyxx}; \quad f_{,yyx}f_{,y}; \quad f_{,yy}; \quad f_{,yxxx}f^5_{,xx}; \quad f^3_{,yx}; \quad f_{,yx}f^5_{,y}; \quad f$$

A more detailed description is given in Section 2. The algorithm itself instructs how to derive, multiply, and substract the two temporary easiest equations in such a way that the most difficult term drops out and a still easier equation can replace the more difficult one, as is also described in Section 2. In Section 3 two examples are given. The stepwise procedure ends if an equation is obtained which contains neither f nor derivatives of f. If this equation is a condition for the variables, then no solution f exists. On the other hand, if the last equation is only the identity $0=0$, then the next-to-last equation may represent a "key equation," meaning that an equation of lower order which is suitably differentiated and multiplied would yield the starting system. For instance,

$$0 = xf_{,x} + 5$$

is a key equation for

$$0 = (xf_{,x} + 5)_{,x} + f(xf_{,x} + 5).$$

A full explanation of what conclusions can be drawn if the algorithm is finished is given in Section 4. While Section 5 concerns a gedankenexperiment dealing with a possible decoupling of the vacuum field equations of general relativity, in Section 6 the computer programme corresponding to the algorithm is described. The characteristic given above makes the method quite applicable for solving Killing equations. In general relativity these are ten equations for four functions whose integrability is not guaranteed.

2. The Algorithm

2.1. *General Properties*

As a demonstration of the method, we shall deal with two equations for one function. This example can be easily generalized to m equations for n functions, see, e.g., the computer programme SPLIT in Section 6. The procedure can be divided into three parts, namely for algebraic, ordinary differential, and partial differential equations. Each part makes frequent use of the previous ones. In the general case, the equations must be given as sums of products of elementary functions of the variables and of products of arbitrary partial derivatives of the functions to be decoupled with positive integer exponents. For instance $5\sin(x)f_{,x}f^3_{,yyx}$ may occur but $\sin(f)$ is not allowed. Further it is assumed throughout that all functions are sufficiently often differentiable. Because the execution involves the multiplication and differentiation of equations (for instance, after multiplication of an equation with f, a new solution is $f=0$), all results obtained are necessary.

2.2. *The Method for Algebraic Equations*

The first part is essentially the Euclidean algorithm for the determination of the greatest common factor of two polynomials. If the given equations for determining f are algebraic, that means

$$P_1 := a_n f^n + \cdots + a_0 = 0 \tag{1a}$$

$$P_2 := b_m f^m + \cdots + b_0 = 0, \qquad m \leqslant n,\ a_n b_m \neq 0 \tag{1b}$$

then the more difficult one $P_1 = 0$ can be replaced by the easier equation $P_3 = b_m P_1 - a_n f^{n-m} P_2$, because the degree of P_3 is always smaller than n. If P_3 is not identical to zero then $P_2 = 0$ and $P_3 = 0$ are combined so that an equation with a still smaller degree is obtained and so on. The procedure is completed if a P_r is reached, which does not contain any powers of f.

EXAMPLE 1.

$$0 = f^3 - 4f^2 + 5f - 2 = P_1\ (= (f-1)^2(f-2))$$

$$0 = 3f^2 - f - 2 = P_2 \qquad (= 3(f-1)^2 + 5(f-1)).$$

ALGORITHM.

$$\begin{aligned} 0 &= 3(f^3 - 4f^2 + 5f - 2) \\ &\quad - f(3f^2 - f - 2) \\ &= 3(-11f^2 + 17f - 6) \\ &\quad + 11(3f^2 - f - 2) \\ &= 3f(40f - 40) \\ &\quad - 40(3f^2 - f - 2) \\ &= -80f + 80 \\ &\quad + 2(40f - 40) \\ &= 0 \end{aligned}$$

The result is, that $0 = 40f - 40$ is a necessary and, as a test also shows, sufficient condition for $P_1 = 0$ and $P_2 = 0$.

2.3. *The Method for Ordinary Differential Equations (ODE)*

If one has two ODEs for one function $f(x)$,

$$D_1 := a_{nk}(f^{(k)})^n + \cdots + a_{1k} f^{(k)} + a_{pk-1}(f^{(k-1)})^p + \cdots + \cdots + a_{00} = 0 \tag{2a}$$

$$D_2 := b_{ml}(f^{(l)})^m + \cdots + b_{00} = 0 \tag{2b}$$

where $f^{(k)}$ and $f^{(l)}$ are the highest derivatives of f, and n, m are the highest exponents of $f^{(k)}$ respectively $f^{(l)}$; a_{nk} and b_{ml} are sums of products of elementary functions of the variables of f and derivatives of f of an order smaller than k and l, respectively; e.g., $a_{nk} = e^x(f^{(k-1)})^{n+5}$.

Case I. $k > l$. Take $D_1 = 0$ and $D_2^{(k-1)} = 0$ as the system to be solved and proceed as in Case II to get an equation with less derivatives than k, and substitute this equation for $D_1 = 0$.

Case II. $k = 1$. Take the system as two polynomials in $f^{(k)}$, use the algorithm given in Section 2.2 either to find the polynomial of lowest order in $f^{(k)}$ which is compatible with both equations and end, or to find an equation without $f^{(k)}$ which therefore has less derivatives than k.

EXAMPLE 2.

$$f = f(x),\ 0 = f'' + f'f + fx + 1 = D_1(= (f' + x)' + (f' + x)f)$$
$$0 = f'^2 + 2xf' + 3f' + x^2 + 3x = D_2(= (f' + x)^2 + 3(f' + x)).$$

ALGORITHM.

$$\begin{aligned} 0 &= (f'^2 + 2xf' + 3f' + x^2 + 3x)' \\ &\quad - (2f' + 2x + 3)(f'' + f'f + fx + 1) \\ &= -2f'^2f - 4f'fx - 2x^2f - 3f'f - 3xf \\ &\quad + 2f(f'^2 + 2xf' + 3f' + x^2 + 3x) \\ &= f'(3ff' + 3fx) \\ &\quad - 3f(f'^2 + 2xf' + 3f' + x^2 + 3x) \\ &= 3xff' + 9ff' + 9fx + 3fx^2 \\ &\quad - (x + 3)(3ff' + 3fx) \\ &= 0. \end{aligned}$$

The result is that $0 = 3ff' + 3fx$ is a necessary condition and as a test shows that $f = 0$ leads to a contradiction, whereas $D_1 = 0 = D_2$ is satisfied by $0 = f' + x$.

2.4. *The Method for Partial Differential Equations (PDE) (the Actual Problem)*

For PDEs an order of the variables must be defined; e.g., $f(y, x)$ is to be decoupled and y is the first and x the second variable. The two equations are

$$\begin{aligned} D_1 := a_{nkr}\left(\frac{\partial^{r+k}}{\partial y^r \partial x^k} f\right)^n + \cdots + a_{1kr}\frac{\partial^{r+k}}{\partial y^r \partial x^k} f \\ + a_{pk-1r}\left(\frac{\partial^{r+k-1}}{\partial y^r \partial x^{k-1}} f\right)^p + \cdots + a_{qvr-1}\left(\frac{\partial^{v+r-1}}{\partial y^{r-1} \partial x^v} f\right)^q + \cdots + a_{000} = 0 \end{aligned} \tag{3a}$$

$$D_2 := b_{mls}\left(\frac{\partial^{s+l}}{\partial y^s \partial x^l} f\right)^m + \cdots + b_{000} = 0 \tag{3b}$$

and n, k, r, m, l, s have the following meaning: $\partial^r f/\partial y^r$, $\partial^s f/\partial y^s$ are the highest y derivatives, and $(\partial^{r+k}/\partial y^r \partial x^k) f$, $(\partial^{s+l}/\partial y^s \partial x^l) f$ are the highest x derivatives of $(\partial^r/\partial y^r) f$ and $(\partial^s/\partial y^s) f$, respectively; n, m are the highest exponents of $(\partial^{r+k}/\partial y^r \partial x^k) f$ and $(\partial^{s+l}/\partial y^s \partial x^l) f$, respectively. a_{nkr} and b_{mls} are sums of products of elementary functions of the variables and of f and its partial derivatives with fewer y derivatives than r and s, respectively, or with y derivatives of the rth or sth order with fewer x derivatives than k and l, respectively. E.g.,

$$a_{nkr} = e^x \frac{\partial^{r+1+k}}{\partial y^{r-1} \partial x^{k+2}} f + \frac{1}{y} \left(\frac{\partial^{r+k-1}}{\partial y^r \partial x^{k-1}} f \right)^{n+5}.$$

Case I. $r > s$. Take $D_1 = 0$ and $(\partial^{r-s}/\partial y^{r-s}) D_2 = 0$ as the system to be solved and proceed as in Case II to get an equation with fewer y derivatives than r and substitute this for $D_1 = 0$.

Case II. $r = s$. The trick is to consider $D_1 = 0$ and $D_2 = 0$ as a system of ordinary differential equations for the function $\partial^r f/\partial y^r$ and the variable x and proceed as in 2.3 until $\partial^r f/\partial y^r$ is eliminated. Because under 2.3 only x-derivatives are performed, the algorithm is finite, and a repeated application of the Cases I and II of 2.4 provides two equations without y derivatives which are decoupled by 2.3. If derivatives of more than two different variables occur, then the calculation runs analogously but the calculational expense grows rapidly. How the method works as a whole will be shown in the following two examples.

3. Two Examples

Example 4. $f(y, x)$ is to be decoupled; y is the first and x is the second variable. The algorithm is shown in Table I. Starting system:

$$D_1 := f + f_{,yy} f_{,x} = 0$$

$$D_2 := f_{,y} + f_{,x}^2 = 0.$$

So $f = 0$ is a necessary and, as a short consideration shows, also a sufficient condition for

$$f + f_{,yy} f_{,x} = 0$$

and

$$f_{,y} + f_{,x}^2 = 0.$$

Example 5. This example deals with a problem arising from the intention to find solutions of the field equations of general relativity. The example can in principle be solved by the methods described in 2.2–2.4, but it is too difficult to solve in

TABLE I

Result of Example 4

Algorithm	Comments
$D_1 := D_1 - D_{2,y} f_{,x} = f - 2f_{,x}^2 f_{,yx} = 0$	D_1 is replaced $\rightarrow$ 2 equations with only one y derivative.
$D_1 := D_1 + D_{2,x} 2f_{,x}^2 = f + 4f_{,xx} f_{,x}^3 = 0$	D_1 is replaced $\rightarrow$ one first equation without y derivative.
$D_3 := D_{2,xx} 4f_{,x}^3 - D_{1,y} = 8f_{,xx}^2 f_{,x}^3 + 8f_{,xxx} f_{,x}^4 - 12f_{,xx} f_{,x}^2 f_{,yx} - f_{,y} = 0$	D_3 is a provisional equation with the highest derivative $f_{,yx}$.
$D_3 := D_3 + D_{2,x} 12f_{,xx} f_{,x}^2 = 32f_{,xx}^2 f_{,x}^3 + 8f_{,xxx} f_{,x}^4 - f_{,y} = 0$	D_3 is replaced and has now $f_{,y}$ as most "difficult" derivative.
$D_2 := D_2 + D_3 = 32f_{,xx}^2 f_{,x}^3 + 8f_{,xxx} f_{,x}^4 + f_{,x}^2 = 0$	D_2 is replaced by a second equation without y derivative.
$D_2 := D_{1,x} 2f_{,x} - D_2 = -8f_{,xx}^2 f_{,x}^3 + f_{,x}^2 = 0$	D_2 is replaced by an equation with only two derivatives of x.
$D_2 := D_2 + D_1 2f_{,xx} = 2ff_{,xx} + f_{,x}^2 = 0$	D_2 is now linear in $f_{,xx}$.
$D_2 := -D_2 2f_{,x}^3 + D_1 f = f^2 - 2f_{,x}^5 = 0$	D_2 is of first order in x.
$D_1 := D_{2,x} 2 + D_1 5f_{,x} = 9f_{,x} f = 0$	D_1 is a second expression of first order in x.
$D_2 := D_1 2f_{,x}^4 + D_2 9f = 9f^3 = 0$	D_2 is a polynomial.
$D_1 := D_{2,x} - D_1 3f = 0 = 0$	The algorithm finishes.

practice, (i.e., with the computer available to the author). Therefore only a list of the performed calculations will be given. Only the number of terms and the "most difficult" derivative of $f(u, x, y, v)$ are written down. This derivative is obtained by taking the highest u derivative and then by choosing from these terms the highest x derivative, and from the remaining terms the highest y derivative, from which the highest v derivative is taken. In Table II $f = f(u, x, y, v)$ is to be decoupled; $Q = Q(x)$ is undetermined. Starting system:

TABLE II

Result of Example 5

Terms	Most difficult derivative	Terms	Most difficult derivative
7	$f_{,u}$	78	$f_{,xxx}$
7	$f_{,xx}$	76	$f_{,xxy}$
26	$f_{,uxv}$	71	$f_{,xxv}$
36	$f_{,ux}$	45	$f_{,xx}$
40	$f_{,uyy}$	48	$f_{,xx}$
45	$f_{,uyv}$	34	$f_{,xy}$
49	$f_{,uy}$	134	$f_{,xxv}$
50	$f_{,uvv}$	142	$f_{,xx}$
68	$f_{,uv}$	152	$f_{,xyy}$
85	$f_{,u}$	581	$f_{,xyv}$

$$-f_{,uv}\frac{u}{x^2}-f_{,v}\frac{1}{2x^2}+e^{2v}Qu^2(2f_{,v}+f_{,vv})+f_{,xx}+f_{,yy}=0$$

$$-f_{,v}f_{,u}\frac{u}{x^2}+e^{2v}f_{,v}^2Qu^2+f_{,x}^2+f_{,y}^2=0.$$

These examples as well as other applications show that the length of the expressions need not grow exponentially over many steps and that the algorithm has more than theoretical value.

4. Possible Results

The algorithm ends if an equation is obtained that contains neither f nor derivatives of f. If this equation is not an identity $0=0$, then either no solution f of the starting system exists, or a condition for other functions (coefficients) appearing in the starting system has to be satisfied as the example shows:

Examples 6, 7.

$$f(x), g(x) \qquad \left.\begin{array}{l} 0=f'+f+x+4 \\ 0=f'+f+6 \end{array}\right\} \quad 0=x-2 \rightarrow \nexists \text{ solution}$$

$$\left.\begin{array}{l} 0=f'+f+x+4 \\ 0=f'+f+g \end{array}\right\} \quad 0=g-x-4 \rightarrow \text{ equation for } g.$$

On the other hand, if the algorithm stops because $0=0$ is obtained, then the next-to-last equation may ba a "key equation" (like the equation $f'+x=0$ in Example 2). But even if $0=0$ results after the first step the method gives hints for further integration as can be seen in Example 8.

Example 8. $f=f(x, y)$:

$$D_1 := 2f_{,xx}^2+2f_{,x}f_{,xxx}+f_{,yxx}f_{,xx}+2f_{,yx}f_{,xxx}+f_{,y}f_{,xxxx}=0$$

$$D_2 := 2f_{,x}f_{,xy}+f_{,yy}f_{,xx}+f_{,y}f_{,xxy}=0.$$

Because $D_{1,y}-D_{2,xx}=0$, no new equation is generated. But the new knowledge $D_{1,y}=D_{2,xx}$ demonstrates the existence of a potential V with $D_1=V_{,xx}$ and $D_2=V_{,y}$. The original information $D_1=0$ and $D_2=0$ yields $V=ax+b$, a, b constant. Integration of $D_2=0$ gives

$$0=f_{,x}^2+f_{,y}f_{,xx}+ax+b$$

as the "key equation" of the starting system. This principle also works if, e.g., $g(x)\,D_{1,y} - D_{2,x} = 0$ but not, if for an arbitrary $g(x, y)$, the relation $g(x, y)\,D_{1,y} - D_{2,x} = 0$ ends the algorithm.

5. The Vacuum Field Equations of General Relativity

If one would start to decouple the ten vacuum field equations $0 = R_{ab}$ of general relativity, the algorithm would yield the contracted Bianchi identities

$$0 = G^{ab}_{,b} + (\Gamma^a_{mn} + \Gamma^b_{bn}\delta^a_m)\, G_{mn} \tag{4}$$

with

$$G^{ab} = R^{ab} - \frac{1}{2} g^{ab} R. \tag{5}$$

In its most general form Eq. (4) does not give any utilizable information, but if the metric has a Killing vector ξ_a with

$$\xi_{a;b} + \xi_{b;a} = 0 \tag{6}$$

then $(-g)^{1/2}_{,b} = (-g)^{1/2}\,\Gamma^a_{ab}$ and Eq. (4) provides that

$$\begin{aligned} 0 &= (-g)^{1/2}\,\xi_a G^{ab}_{,b} + (-g)^{1/2}\,\xi_a(\Gamma^a_{mn} + \Gamma^b_{bn}\delta^a_m)\, G^{mn} \\ &= ((-g)^{1/2}\,\xi_a G^{ab})_{,b}. \end{aligned} \tag{7}$$

With $((-g)^{1/2}\,\xi^a R)_{,a} = (-g)^{1/2}\,\xi^a_{;a} R + (-g)^{1/2}\,\xi^a R_{;a} = 0$, Eq. (7) reads

$$0 = ((-g)^{1/2}\,\xi^a R^b_a)_{,b}. \tag{8}$$

Indeed, $(-g)^{1/2}\,\xi^a R^b_a$ is expressible as a curl:

$$(-g)^{1/2}\,\xi^a R^b_a = -(\xi^{[b;c]}(-g)^{1/2})_{,c}. \tag{9}$$

To integrate Eq. (9) for $R^b_a = 0$ by $-\xi^{a;b}(-g)^{1/2} = \varepsilon^{abcd} M_{c,d}$ is not the best answer because not all M_c are relevant. By taking into consideration the natural $3+1$ splitting induced by ξ^a, assured of an analogous relation as Eq. (9) in $3+1$ formulation, and by introducing the projection tensor h_{ab}, the Levi-Civita tensor ε_{abc} (both tensors on the 3 space Σ_3, the factor space V_4/G_1), and the twist vector ω^a, where

$$\begin{aligned} &h_{ab} = g_{ab} + u_a u_b,\; u^a \equiv (-F)^{-1/2}\,\xi^a, \qquad F = \xi_a \xi^a, \\ &\varepsilon_{abc} = \varepsilon_{dabc} u^d, \qquad \omega^a = \varepsilon^{abcd}\xi_b \xi_{c;d}, \end{aligned} \tag{10}$$

one finds that

$$\omega_{[b\|a]} = \varepsilon_{abmn}\,\xi^m R^n_d\,\xi^d \tag{11}$$

$$(-F)^{-1/2}\,\varepsilon^{abc}\omega_{c\|b} = 2h^a_b R^b_c\xi^c. \tag{12}$$

ξ^a is assumed to be timelike, while only the signs must be changed. The conclusion for vacuum is now $\omega_c = \omega_{,c}$. The existence of the twist potential ω was already proved in [2].

With the complex potential $\Gamma = F + i\omega$ and the conform transformed metric $\gamma_{ab} = -Fh_{ab}$ the field equations finally read

$$\hat{R}_{ab} = \tfrac{1}{2}F^{-2}\Gamma_{,(a}\bar{\Gamma}_{,b)}, \qquad \Gamma^{;a}_{,a} + F^{-1}\gamma^{ab}\Gamma_{,a}\Gamma_{,b} = 0,$$

where $\hat{R}_{ab}$ is the Ricci tensor and ; is the covariant derivative with respect to γ_{ab}. For more detail see [1]. It must be mentioned that for the case of a homothetic vector (8) holds, but the $3+1$ splitting does not work to introduce a twist potential. These remarks should be an example of how the information of the structure or the explicit form of the integrability conditions may help find new potentials and formulations of the equations to be solved.

6. The Corresponding Computer Programme SPLIT

The algorithm was implemented in FORMAC 73 because other systems like REDUCE II are more expensive in core and time. To run SPLIT, all functions in the order to be decoupled, all equations, and the number of equations must be given. The order of the arguments within the functions determines the sequence of the variables. For instance, to run Example 4, the following instructions are necessary:

```
SPLIT: PROC OPTIONS(MAIN);
       FORMAC OPTIONS;
       LET(F#(1)=F.(Y,X);
           D#(1)=F.(Y,X)+DERIV(F.(Y,X),Y,2)*DERIV(F.(Y,X),X);
           D#(2)=DERIV(F.(Y,X),Y)+DERIV(F.(Y,X),X)**2);
       LL# =2; CALL PTWSEP;
PTWSEP: PROC;
           :
         END PTWSEP;
         END SPLIT;
```

Here the two (because LL# =2) equations D#(1)=0, D#(2)=0 are to be decoupled with respect to the one function F(Y,X). Thereby the variable Y is the first one with the consequences mentioned in Section 2.4. If m equations and n functions should be decoupled, then after some preliminary equations $m-1$

equations without F#(1) would be generated, and out of them $m-2$ equations without F#(2) would be generated and so on. Within this sequence every equation is characterized by a number according to the highest order derivatives of the function to be removed instantaneously, i.e., according to their degree of "difficulty." The two equations with the highest numbers are combined by the subroutine PTWSPL, respectively, PTWNSE if the equations are algebraic, and every new equation is analysed by a subroutine PTWSEA until one equation with a lower number is generated, which will replace the more difficult of the two initial equations. Again the two equations with the highest numbers are combined. After repeated execution, eventually the function in question occurs in only one equation, and the next function together with the residual equations is taken.

The possibilities of the programme become apparent in Example 5. Up to 581 terms, the calculation was performed on a computer EC1040 without virtual memory, comparable to the IBM 360, and with a storage capacity of 750 k bytes for the programme and data and with a CPU-time of 220 s. The source code in FORMAC 73 consists of 340 cards. For further information please contact the author.

References

1. D. Kramer, H. Stephani, M. A. H. MacCallum, and E. Herlt, "Exact Solutions of Einstein's Field Equations," VEB Deutscher Verlag der Wissenschaften, Berlin, Cambridge Univ. Press, Cambridge, 1980.
2. A. Papapetrou, "Quelques remarques sur les champs gravitationels stationnaires," *C. R. Acad. Sci.* (Paris) **257** (1963), 2797.